

# AOCC Install Guide

## Contents

- [AOCC Install Guide](#)
  - [Installing AOCC on Linux](#)
  - [Supported Operating systems](#)
  - [Known issues and limitations](#)

“AMD Optimizing C/C++ Compiler” - abbreviated as AOCC is a highly optimized C, C++ and Fortran compiler for x86 targets especially for Zen based AMD processors. It supports Flang as the default Fortran front-end compiler. This manual does not cover the internals of LLVM or AOCC compiler.

This AOCC 2.1 is based on LLVM 9.0 release (llvm.org, 19th Sep 2019) with improved Flang Fortran front-end added with F2008 features and bug fixes

## Software Requirement

AOCC requires below software packages to be installed. The version column indicates the known to work version of these packages. Existence of these prerequisite packages can be checked using our AOCC-prerequisites-check script as mentioned below.

Package	Versions	Notes
GCC	>=5.1.0	C/C++ Compiler
libstdc++	>=6	GNU Standard C++ Library V3
libncurses-dev	>=5.9	Provides libtinfo, a lowlevel terminfo library
Zlib	>=1.2.7	Compression library

Note: Recommend to use the latest versions of Glibc and Binutils for better performance

# Installing AOCC on Linux

## C, C++ and Fortran Compiler

- AOCC package is available in tar, rpm and deb types
- Download the appropriate package to a directory of your choice say <compdir> and follow the below installation steps

- **TAR - Installation does not require root or sudo permission**

- `cd <compdir>`
- `tar -xvf aocc-compiler-2.1.0.tar*`
- `cd aocc-compiler-2.1.0`
- `bash install.sh`
  - This will install the compiler and display AOCC setup instruction
- `source <compdir>/setenv_AOCC.sh`
  - This will set up shell environment for using AOCC C, C++ and Fortran compiler in the shell environment where above command was executed

- **RPM: Installation needs root or sudo permission**

- `cd <compdir>`
- `sudo rpm -ivh aocc-compiler-2.1.0-1.x86_64.rpm`
  - This will install the compiler under /opt/AMD/aocc-compiler-2.1.0
- `source /opt/AMD/aocc-compiler-2.1.0/setenv_AOCC.sh`
  - This will set up shell environment for using AOCC C, C++ and Fortran compiler in the shell environment where above command was executed

- **DEB: Installation needs root or sudo permission**

- `cd <compdir>`
- `dpkg -i aocc-compiler-2.1.0_1_amd64.deb`
  - This will install the compiler under /opt/AMD/aocc-compiler-2.1.0
- `source /opt/AMD/aocc-compiler-2.1.0/setenv_AOCC.sh`
  - This will set up shell environment for using AOCC C, C++ and Fortran compiler in the shell environment where above command was executed

- **“Clang -v” to check the installed AOCC compiler**

- This will show AOCC version

- You can run `bash <compdir>/AOCC-2.1-Compiler/AOCC-2.1-prerequisites-check.sh` to check if you have all the prerequisites and that your shell environment is set up correctly
  - If there are failing checks then correct them (repeat any of the above steps that you may have missed) and run `prerequisites_check.sh` again
  - Repeat until `prerequisites_check.sh` displays 'Check:PASSED'
  - Note: If you have only warnings in failing checks then you can proceed if packages mentioned in the warnings are not required for your run
- The compiler is installed, and your environment set to point to the current release of AOCC. At any point in time you can `source <compdir>/setenv_AOCC.sh` to set the environment variables for the installed compiler

## Using AOCC

- `source <compdir>/setenv_AOCC.sh`
- To build and run a C/C++ program
  - `$ clang [command line options] xyz.c -o xyz`
  - `$ ./xyz`
  - Optimization flags: Please read the options listed in [Clang - the C, C++ Compiler guide](#)
- To build and run Fortran programs using flang:
  - `$ flang [command line options] hello.f90 -o hello`
  - `$ ./hello`
  - Optimization flags: Please read the options listed in [Flang - the Fortran Compiler guide](#)

Some applications may benefit from optimized libraries. AOCC is known to work seamlessly with these libraries. It is recommended that you evaluate these libraries while building your application with AOCC as they may help boost the performance of your application over and above the compiler optimizations that come with AOCC.

In most cases these libraries only need to be linked Example

- `export LD_LIBRARY_PATH=<compdir>/aocc-compiler-2.1.0/lib:$LD_LIBRARY_PATH`
- `clang [command line flags] xyz.c -L<compdir>/aocc-compiler-2.1.0/lib -lamdlibm -o xyz`
- `./xyz`

**Note:** For environment module users, you can find the module file "`aocc-compiler-2.1.0_module`". This support will be available only with tar package.

## Upgrading AMD LibM (Required only when upgrading AMD LibM from [here](#) )

- Extract the newer AMD LibM version package
- Overwrite *AOCC-<ver>-Compiler/lib/libamdlbm.so* and *AOCC-<ver>-Compiler/lib/libamdlbm.a* with the newer versions of *libamdlbm.so* and *libamdlbm.a*
- Similarly, overwrite *AOCC-<ver>-Compiler/include/amdlbm.h* with the newer versions of *amdlbm.h*

## Supported Operating systems

This release has been tested on the following OS

- RHEL 7.4
- SLES 12 SP3
- Ubuntu 18.04 LTS

## Known issues and limitations

- AOCC compiler binaries are suitable to run on Linux systems having glibc version 2.17 and above only.
- Currently, Flang supports only 64 bit targets